3208 Monocolor LED 3mm (5mm) Dot Matrix Display Information Board User's Guide

Table of Contents

NOTES:

Product Version : Ver 1.0

Document Version : Ver 1.0

3208 MONOCOLOR LED 3MM (5MM) DOT MATRIX DISPLAY INFORMATION BOARD USER'S GUIDE

Chapter 1. Overview

1.1 Overview

Thanks for using 3208 monocolor LED dot matrix info board series by Sure Electronics. Integrating HT1632Cs as driver chips, these info boards support 16-level PWM brightness control and all LED dot matrixes displayed are mapped to the RAM of HT1632C. Peripheral circuits are required to light up LEDs via the ports on the boards. These info boards can be used to display digits, letters and even graphs. It is allowed to connect up to 4 boards of the same kind in series for wider applications such as info display in banks, stores, households and so on. You may refer to the following table for members of this series.

TABLE 1-1 3208 MONOCOLOR LED DOT MATRIX DISPLAY INFO BOARD SERIES

Product Number	Product Name
DE-DP13111	3208 Green LED 3mm Dot Matrix Display Information Board
DE-DP13112	3208 Red LED 3mm Dot Matrix Display Information Board
DE-DP13211	3208 Green LED 5mm Dot Matrix Display Information Board
DE-DP13212	3208 Red LED 5mm Dot Matrix Display Information Board

FIGURE 1-1 OVERVIEW

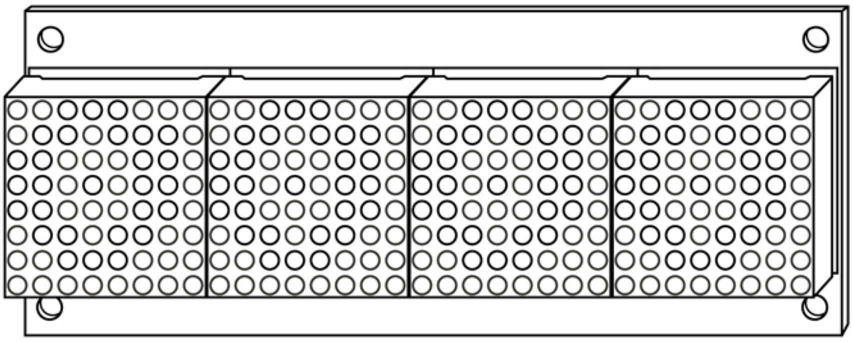


FIGURE 1-2 BACK VIEW OF 3208 LED 3MM DOT MATRIX DISPLAY INFO BOARD

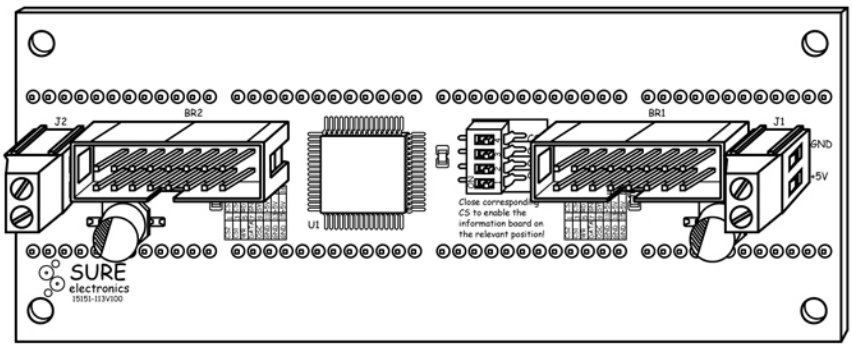
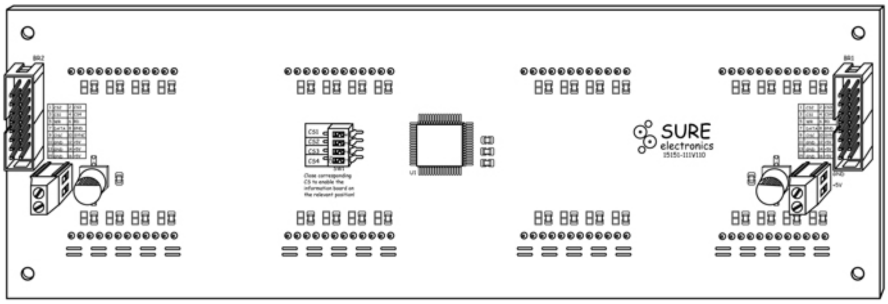


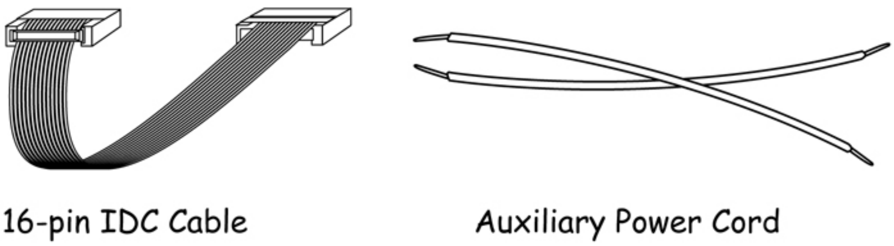
FIGURE 1-3 BACK VIEW OF 3208 LED 5MM DOT MATRIX DISPLAY INFO BOARD



1.2 Quick Start

A 16-pin IDC cable and two power cords are provided for free. DE-DD210 by Sure Electronics is used in this manual as a driver board. Program this driver board to control the display on the info board.

FIGURE 1-4 ACCESSORIES



Note:
1. Other driver board can be used. You may refer to **2.2 Port Definition** to do relative adjustments.
2. Sample codes are provided in this manual for reference.

1.2.1 Connection of One Info Board and The Driver Board

Connect BR1 of the info board and BR1 of the driver board with a 16-pin IDC cable and push CS1 of DIP switch on the info board to ON.

FIGURE 1-5 CONNECTION OF THE DRIVER BOARD AND 3208 MONOCOLOR LED 3MM DOT MATRIX DISPLAY INFO BOARD

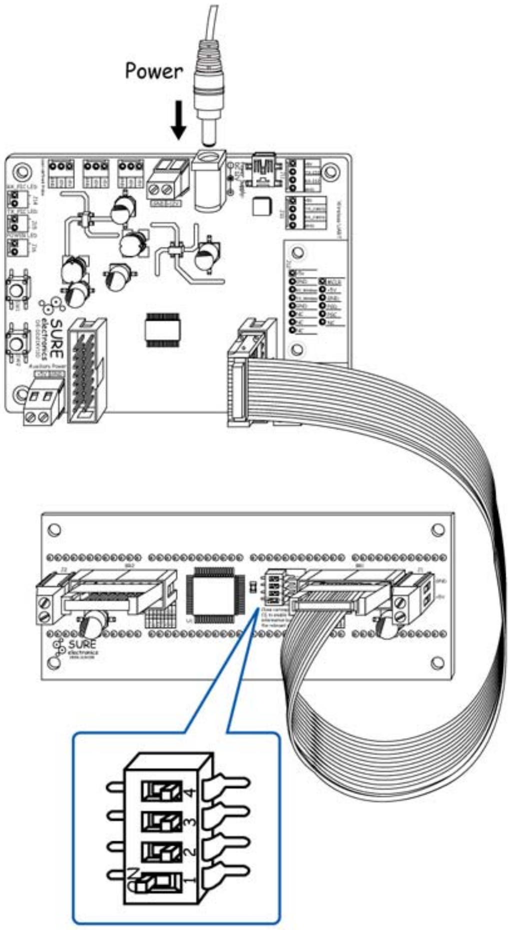
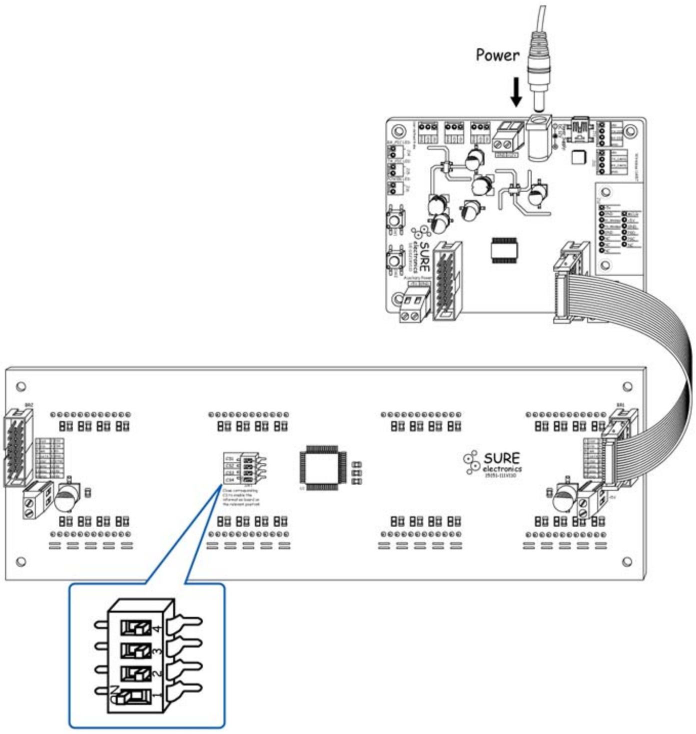


FIGURE 1-6 CONNECTION OF THE DRIVER BOARD AND 3208 MONOCOLOR LED 5MM DOT MATRIX DISPLAY INFO BOARD



Program codes to the chip of the driver board and repower the board.

Note: If you're not familiar with programming, try using the sample codes first.

1.2.2 Connection of Many Info Boards (Max 4 Boards)

First, an auxiliary power supply is recommended to power the info boards connected in series via the auxiliary power terminals: J1 and J2. Connect +5V, GND of J2 on one info board and the corresponding +5V, GND of J1 on the next info board with power cords. The auxiliary supply should be able to output DC5V 1.5A.

Connect BR1 of the driver board and BR1 of the info board with a 16-pin IDC cable. Then, as shown in figure 1-6, connect four 3208 info boards and the driver board with four 16-pin IDC cables. Set the CS1 of SW1 of the first info board, CS2 of the second info board, CS3 of the third info board and CS4 of the fourth info board ON.

FIGURE 1-6 CONNECTION OF MANY 3MM DOT MATRIX INFO BOARDS IN SERIES

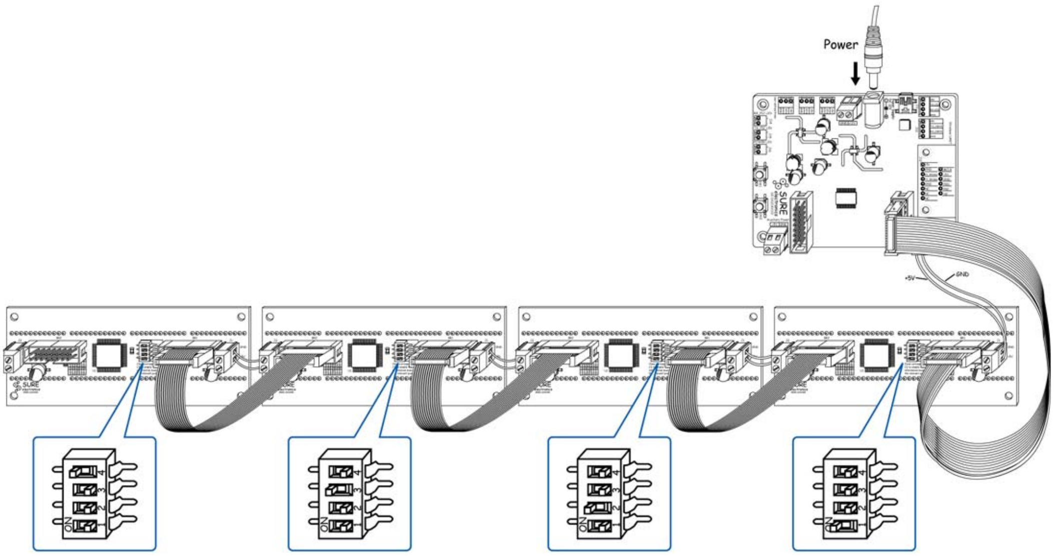
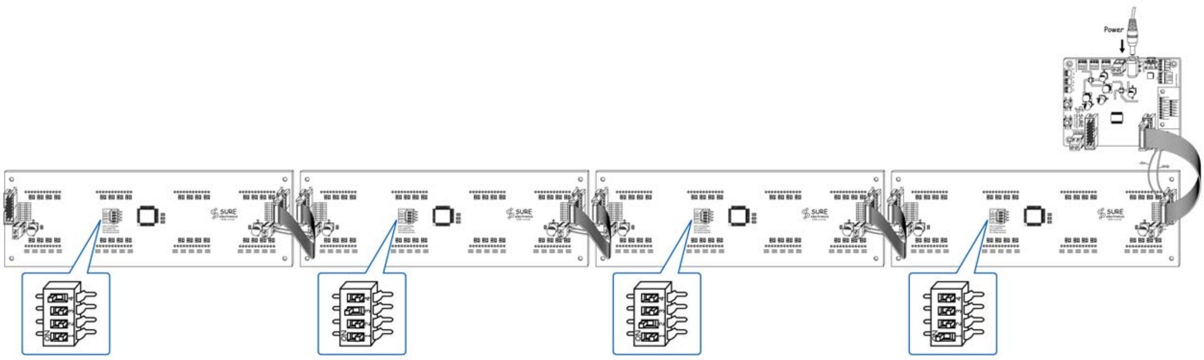


FIGURE 1-7 CONNECTION OF MANY 5MM DOT MATRIX INFO BOARDS IN SERIES



Program the chip on the driver board to control the LED display.

Note: If you use the sample codes, all the boards will display the same content.

Chapter 2. Hardware Detail

2.1 Hardware

1. 4 pieces of 8*8 LED dot matrix display
   Light-emitting diameter of DE-DP13111 and DE-DP13112 is 3mm. Light-emitting diameter of DE-DP13211 and DE-DP13212 is 5mm.
2. LED drive chip (U1): HT1632C, QFP packaging.
3. 16-pin male sockets (BR1 and BR2): used for data, clock, control signal and +5V supply input.
4. Auxiliary power supply terminals (+5V) (J1and J2): for external power input when more info boards are connected in series.

2.2 Port Definition

TABLE 2-1 PIN DEFINITION OF BR1 AND BR2

Pin Number	Pin Name	Function Description
1	CS2	Chip Selection 2
2	CS3	Chip Selection 3
3	CS1	Chip Selection 1
4	CS4	Chip Selection 4
5	WR	WRITE clock input with pull-high resistor Data on the DATA lines are latched into the HT1632C on the rising edge of the WR signal.
6	RD	Read Clock
7	DATA	Data Input
9	OSC	If the RC Master Mode command is programmed, the system clock source is from on-chip RC oscillator and system clock is output to OSC pin. If the Slave Mode or EXT CLK Master Mode command is programmed, the system clock source is input from external clock via the OSC pin
10	SYNC	If the RC Master Mode or EXT CLK Master Mode command is programmed, the synchronous signal is output to SYN pin. If the Slave Mode command is programmed, the synchronous signal is input from SYN pin.
8, 11, 13, 15	GND	GND
12, 14, 16	VCC	Power Supply

2.3 Display Memory

Display is controlled by modifying the data stored in RAM of HT1632C. All LEDs on one board are controlled by only one HT1632C.

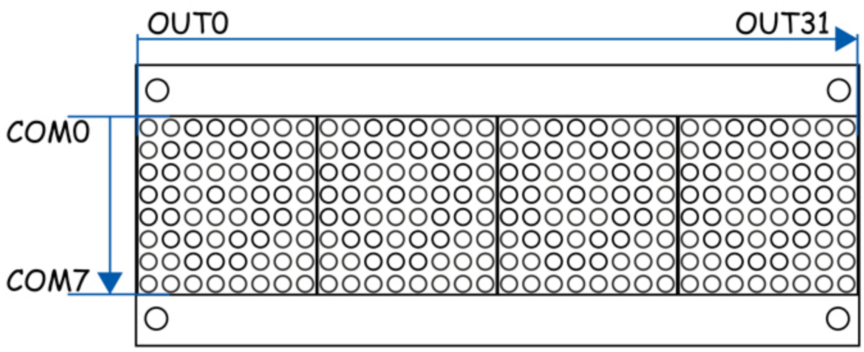
The distribution of HT1632C's corresponding address is shown as follows:

TABLE 2-2 THE CORRESPONDING ADDRESS OF HT1632C

	COM7	COM6	COM5	COM4	Addr	COM3	COM2	COM1	COM0	Addr
OUT0					01H					00H
OUT1					03H					02H
OUT2					05H					04H
OUT3					07H					06H
OUT4					09H					08H
OUT5					0BH					0AH
OUT6					0DH					0CH
OUT7					0FH					0EH
OUT8					11H					10H
OUT9					13H					12H
OUT10					15H					14H
OUT11					17H					16H
OUT12					19H					18H
OUT13					1BH					1AH
OUT14					1DH					1CH
OUT15					1FH					1EH
OUT16					21H					20H
OUT17					23H					22H
OUT18					25H					24H
OUT19					27H					26H
OUT20					29H					28H
OUT21					2BH					2AH
OUT22					2DH					2CH
OUT23					2FH					2EH
OUT24					31H					30H

OUT25					33H					32H
OUT26					35H					34H
OUT27					37H					36H
OUT28					39H					38H
OUT29					3BH					3AH
OUT30					3DH					3CH
OUT31					3FH					3EH
	D7	D6	D5	D4	Data	D3	D2	D1	D0	Data

FIGURE 2-1 THE CORRESPONDING ADDRESS OF HT1632C



2.4 Command Format

CS (CS1、CS2、CS3、CS4) of HT1632C must be set to low before data or command is sent to this HT1632C. When the transmission is complete, CS must be reset to high.

The timing diagram is as follows:

FIGURE 2-2 SEND DATA

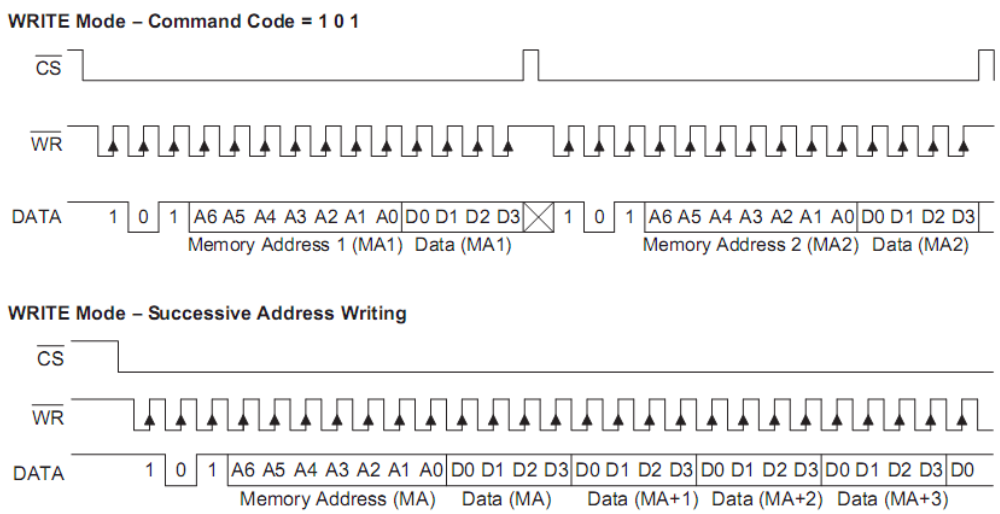
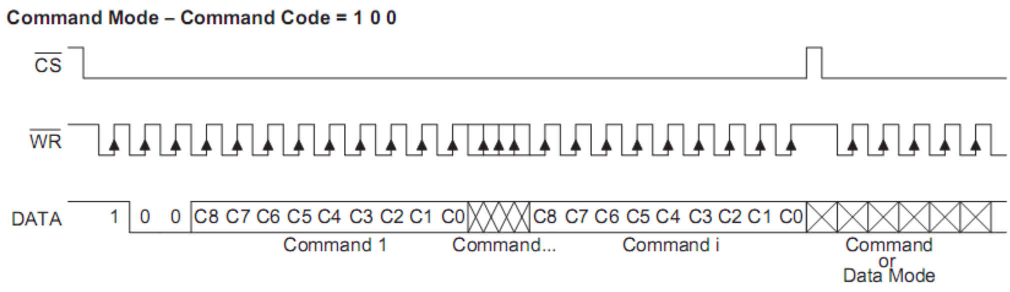


FIGURE 2-3 SEND COMMAND



Note:	You may refer to HT1632C data sheet for details.

2.5 Command Summary

Command summary is shown in the following table. For this info board, these commands - WRITE, SYS EN, LED On, LED Off, RC Master Mode, COM Option and PWM Duty are recommended to be used. Using the commands which are not recommended may cause abnormal display.

FIGURE 2-4 COMMAND SUMMARY

Command Summary

Name	ID	Command Code	D/C	Function	Def.
READ	1 1 0	A6A5A4A3A2A1A0D0D1D2D3	D	Read data from the RAM	
WRITE	1 0 1	A6A5A4A3A2A1A0D0D1D2D3	D	Write data to the RAM	
READ-MODIFY-WRITE	1 0 1	A6A5A4A3A2A1A0D0D1D2D3	D	Read and Write data to the RAM	
SYS DIS	1 0 0	0000-0000-X	C	Turn off both system oscillator and LED duty cycle generator	Yes
SYS EN	1 0 0	0000-0001-X	C	Turn on system oscillator	
LED Off	1 0 0	0000-0010-X	C	Turn off LED duty cycle generator	Yes
LED On	1 0 0	0000-0011-X	C	Turn on LED duty cycle generator	
BLINK Off	1 0 0	0000-1000-X	C	Turn off blinking function	Yes
BLINK On	1 0 0	0000-1001-X	C	Turn on blinking function	
SLAVE Mode	1 0 0	0001-0XXX-X	C	Set slave mode and clock source from external clock, the system clock input from OSC pin and synchronous signal input from SYN pin	
RC Master Mode	1 0 0	0001-10XX-X	C	Set master mode and clock source from on-chip RC oscillator, the system clock output to OSC pin and synchronous signal output to SYN pin	Yes
EXT CLK Master Mode	1 0 0	0001-11XX-X	C	Set master mode and clock source from external clock, the system clock input from OSC pin and synchronous signal output to SYN pin	
COM Option	1 0 0	0010-abXX-X	C	ab=00: N-MOS open drain output and 8 COM option ab=01: N-MOS open drain output and 16 COM option ab=10: P-MOS open drain output and 8 COM option ab=11: P-MOS open drain output and 16 COM option	ab =00
PWM Duty	1 0 0	101X-0000-X	C	PWM 1/16 duty	
	1 0 0	101X-0001-X	C	PWM 2/16 duty	
	1 0 0	101X-0010-X	C	PWM 3/16 duty	
	1 0 0	101X-0011-X	C	PWM 4/16 duty	
	1 0 0	101X-0100-X	C	PWM 5/16 duty	
	1 0 0	101X-0101-X	C	PWM 6/16 duty	
	1 0 0	101X-0110-X	C	PWM 7/16 duty	
	1 0 0	101X-0111-X	C	PWM 8/16 duty	
	1 0 0	101X-1000-X	C	PWM 9/16 duty	
	1 0 0	101X-1001-X	C	PWM 10/16 duty	
	1 0 0	101X-1010-X	C	PWM 11/16 duty	
	1 0 0	101X-1011-X	C	PWM 12/16 duty	
	1 0 0	101X-1100-X	C	PWM 13/16 duty	
	1 0 0	101X-1101-X	C	PWM 14/16 duty	
	1 0 0	101X-1110-X	C	PWM 15/16 duty	
	1 0 0	101X-1111-X	C	PWM 16/16 duty	

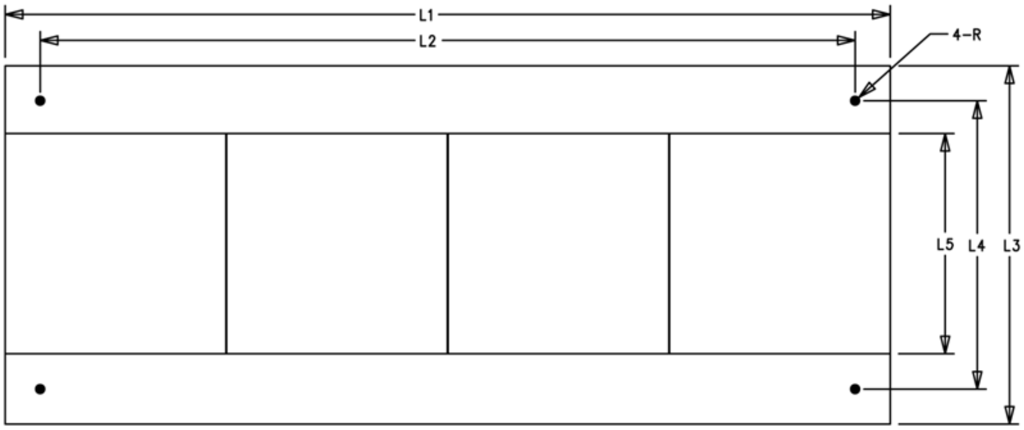
Chapter 3. Electrical Characteristics

FIGURE 3-1 ELECTRICAL CHARACTERISTICS

Parameter	Symbol	Value	Unit
Operating Voltage	V_{in}	5	V
Storage Temperature	T_{stg}	-20 - 80	℃
Average Operating Current	I_{avrg}	0.20	A
Maximum Operating Current (All LEDs on, 100% PWM duty cycle)	I_{max}	0.36	A

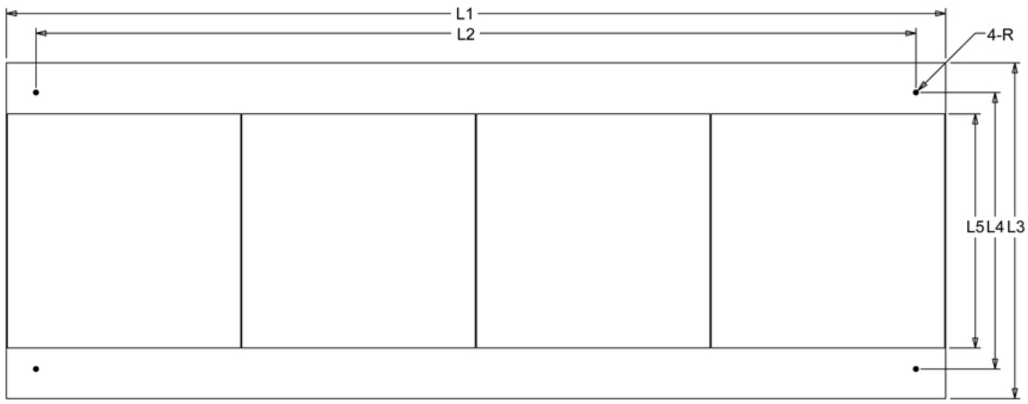
Chapter 4. Mechanical Drawing

FIGURE 4-1 MECHANICAL DRAWING OF ONE 3208 MONOCOLOR LED 3MM DOT MATRIX DISPLAY INFO BOARD



Symbol	L1	L2	L3	L4	L5	R
inch	5.05	4.65	2.05	1.65	1.26	0.06
mm	128.27	118.11	52.07	41.91	32.00	1.52

FIGURE 4-2 MECHANICAL DRAWING OF ONE 3208 MONOCOLOR LED 5MM DOT MATRIX DISPLAY INFO BOARD

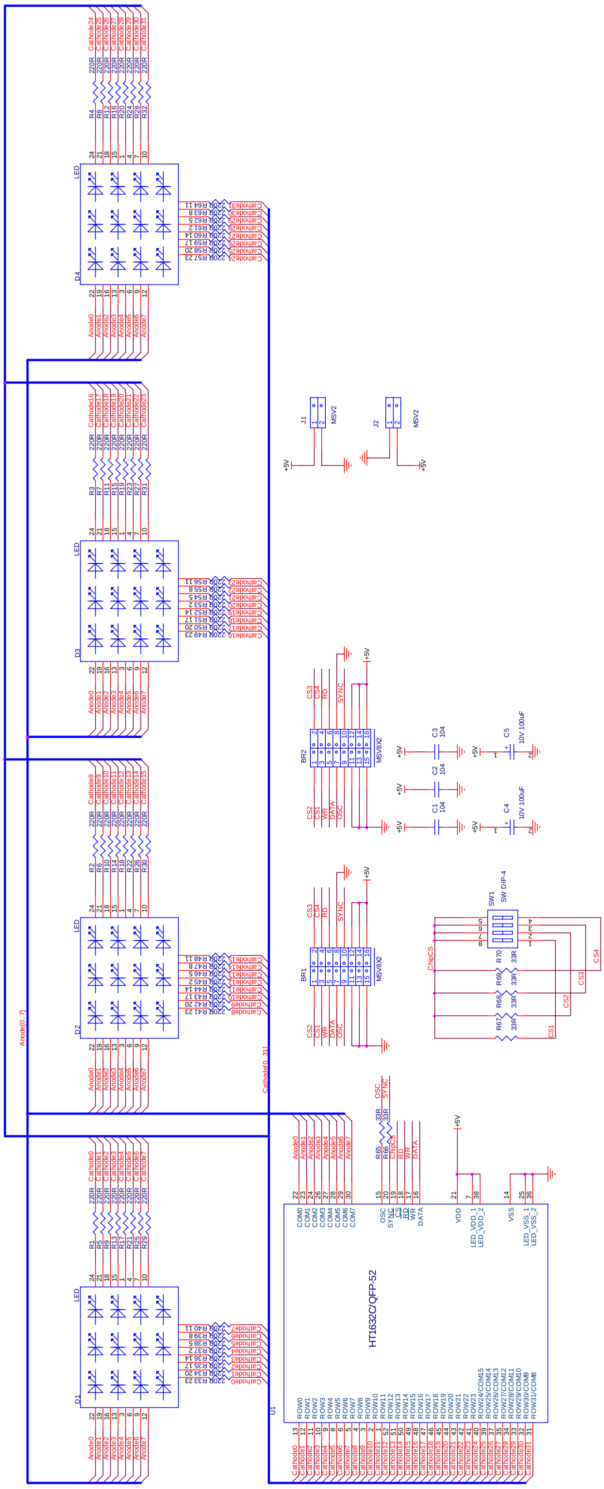


Symbol	L1	L2	L3	L4	L5	R
inch	9.50	8.90	3.40	2.80	2.37	0.06
mm	241.30	226.06	86.36	71.12	60.20	1.52

Chapter 5. Appendix

5.1 Schematic

FIGURE 5-1 SCHEMATIC



5.2 Sample Code

The driver board DE-DD210, integrating PIC16F723 as its master chip, is used as an example. This sample code is used to illuminate the odd rows of LEDs.

Compilation environment: MPLAB IDE v8.40

Compiler: HI-TECH ANSI C Compiler PRO 9.65

File “Declare.h”

```
#ifndef    _DECLARE_
#define    _DECLARE_

//Macro definition of ports used
#define         CS1        RB0        //8 control ports
#define         CS2        RB1
#define         CS3        RB2
#define         CS4        RB3
#define         CS5        RB4
#define         CS6        RB5
#define         CS7        RB6
#define         CS8        RB7
#define         CS_OFF CS1=1;CS2=1;CS3=1;CS4=1;CS5=1;CS6=1;CS7=1;CS8=1;
#define         CS_ON   CS1=0;CS2=0;CS3=0;CS4=0;CS5=0;CS6=0;CS7=0;CS8=0;

#define         CLK        RC3              //Clock line simulating SPI communication (this
                                            //port is also the clock line of SPI communication
                                            //integrated by MCU)
#define         DAT        RC5              //Data line simulating SPI communication (this
                                            //port is also the data line of SPI communication
                                            //integrated by MCU)
#define         SW1        RC0              //Two switches
#define         SW2        RC1

//Following is the functions defined in a way of macro definition.
#define CLK_DELAY;    NOP()

//Following definition facilitates compilation of HT1632C control commands.
#define RC_MASTER_MODE     0b100000110000  //Set master mode and clock
                                           //source from on-chip RC oscillator
#define SYS_DIS            0b100000000000  //Turn off both system oscillator and
                                           //LED duty cycle generator
#define SYS_EN             0b100000000010  //Turn on system oscillator
#define LED_OFF            0b100000000100  //Turn off LED duty cycle generator
#define LED_ON             0b100000000110  //Turn on LED duty cycle generator
#define N_MOS_COM8         0b100001000000  //N-MOS open drain output and 8
                                           //common option
#define PWM_16             0b100101011110  //PWM 16/16 duty

#endif
```

File “SampleCode.c”

```
#include <pic.h>
#include "Declare.h"

//*********************************************************************************************
//Function Name: device file configuration
//Function Feature:  configure MCU's working modes and status
//Input Argument: INTIO: INTOSCIO- internal oscillator, OSC1 and OSC2 used as I/O
//ports
//              WDTDIS: Disable watchdog timer
//              PWRTDIS: Disable power-delay timer
//              MCLREN: Enable MCLR
//              PROTECT: Do NOT protect the code
//              BORDIS: Brown out reset disable
//              BORV25: Brown-out reset voltage set to 2.5V nominal
//              PLLEN:
//              DEBUGEN: In-circuit debugger enabled
//
//              VCAPDIS: Voltage regulator capacitor disable
//Output Argument: void
//*********************************************************************************************
__CONFIG(INTIO & WDTDIS & PWRTEN & MCLREN & UNPROTECT & BORDIS &
BORV25 & PLLEN & DEBUGEN);
__CONFIG(VCAPDIS);

//Function Prototype Declaration
void SystemInit(void);                      //System Initialization
void SetHT1632CAs3208(void);                //Set HT1632C to work in 32*8 mode

void CommandWrite(unsigned int command);    //Write commands to all HT1632Cs
void AddressWrite(unsigned char address);   //Write address

void SPI_ModelConfigure(void);
void SPI_DataSend(const unsigned char data);
void PrintChar(unsigned char c);            //Transmit single char
void Print(void);

void main()
{
    SystemInit();
    SetHT1632CAs3208();
    CS_ON;
    Print();
    CS_OFF;
    while(1)        //Stay here forever
    {
```

```
        ;
    }
}

//***************************************************************************************************
//Function Name: system initialization
//Function Feature: set corresponding data reading and writing of PORTB and PORTC
//Input Argument: void
//Output Argument: void
//***************************************************************************************************
void SystemInit(void)
{
    IRCF1 = 1;          //Set the frequency of the internal oscillator as 8MHz
    IRCF0 = 0;
    BRGH=0;             //Select low baud rate mode, default status after power-on reset
    OSCTUNE = 0x1f;     //Oscillator at the maximum frequency
    ANSELB = 0x00;      //PORTB as a digital I/O port
    TRISB = 0x00;       //PORTB as an output port
    PORTB = 0x00;       //Clear PORTB output
    TRISC0 = 1;         //PORTC0 (SW1 port) as an input port
    TRISC1 = 1;         //PORTC1 (SW2 port) as an input port
    TRISC3 = 0;         //PORTC3 (CLK signal) as an output port
    TRISC5 = 0;         //PORTC5 (DATA signal) as an output port
    T0IE = 0;           //Turn off interruption of timer0
}

//***************************************************************************************************
//Function Name: SetHT1632C_As3208
//Function Feature: write basic configuration to HT1632C in command words
//Input Argument: void
//Output Argument: void
//***************************************************************************************************
void SetHT1632CAs3208(void)
{
    CommandWrite(SYS_EN);           //Enable system oscillator
    CommandWrite(LED_ON);           //Turn on LED
    CommandWrite(RC_MASTER_MODE);   //Select on-chip RC as the system clock
                                    //working in master mode
    CommandWrite(N_MOS_COM8);       //N-MOS open-drain output and 32 ROW * 8
                                    //COM
    CommandWrite(PWM_16);           //Set the grade of initial PWM brightness as
                                    //light_degree (16/16)
}

//***************************************************************************************************
//Function Name: CommandWrite
```

```
//Function Feature: write control commands to all HT1632Cs
//Input Argument: command words written to "command", specifically stated in "declare"
//function
//Output Argument: void
//Argument Description: compile control commands to all external HT1632Cs for the
//requirement of the project
//***************************************************************************************************
void CommandWrite(unsigned int command)
{
    unsigned char i;
    unsigned int j;
    command = command & 0x0fff;   //12-bit command word, upper four bits masked

    CS_OFF;                   //Disable all HT1632Cs
    CLK_DELAY;
    CS_ON                     //Enable all HT1632Cs
    CLK_DELAY;
    for(i=0; i<12; i++)       //Write command words in HI1632C register
    {
        CLK = 0;
        CLK_DELAY;
        j = command & 0x0800;          //Return the MSB
        command = command << 1;        //Move the control character to the left one
        j = j >> 11;                   //Position the value at the LSB
        DAT = j;                       //Send the value to the data port
        CLK_DELAY;
        CLK = 1;                       //Data transmission (data valid on rising edge)
        CLK_DELAY;
    }
    CS_OFF;                            //Disable all HT1632Cs
}

//***************************************************************************************************
//Function Name: AddressWrite
//Function Feature: write start address of data to HT1632C
//Input Argument: address: address to be written
//Output Argument: void
//***************************************************************************************************
void AddressWrite(unsigned char address)
{
    unsigned char i,temp;
    SSPCON = 0x11;
    address = address & 0x7f;      //7-bit address, mask the MSB
    CLK = 0;                       //Clock line is 0
    CLK_DELAY;
    DAT = 1;                       //Send "1" to data port
```



```
        CLK_DELAY;
        CLK = 1;                      //Data transmission
        CLK_DELAY;
        CLK = 0;
        CLK_DELAY;
        DAT = 0;                      //Send “0” to data port
        CLK_DELAY;
        CLK = 1;                      //Data transmission
        CLK_DELAY;
        CLK = 0;
        CLK_DELAY;
        DAT = 1;                      //Send “1” to data port
        CLK_DELAY;
        CLK = 1;                      //Data transmission
        CLK_DELAY;
        for(i=0; i<7; i++)            //Write “address” to HT1632C register
        {
            CLK = 0;                  //Clock line is 0
            CLK_DELAY;
            temp = address & 0x40;    //Return the MSB
            address = address << 1;   //Shift left once
            temp = temp >> 6;         //Position the value at the LSB
            DAT = temp;               //Send the value to the data port
            CLK_DELAY;
            CLK = 1;                  //Data transmission
            CLK_DELAY;
        }
}

//*********************************************************************************************
//Function Name: SPI_ModelConfigure
//Function Feature: configure the corresponding data transfer port of PIC microcontroller
//for SPI communication
//Input Argument: void
//Output Argument: void
//*********************************************************************************************
void SPI_ModelConfigure(void)
{
        SSPIF = 0;          //Initial state: waiting to send data
        SSPCON = 0x31;      //Write in this register: SSPEN=1 (enable serial port); CKP=1
                            //(CLK high in an idle state); CLK is FOSC/16
        SSPSTAT = 0x80;     //Write in this register: SMP=1(Input data sampled at end of
                            //data output time); CKE=0(data stable on rising edge of SCK)
}

//*********************************************************************************************
```

```
//Function Name: SPI_DataSend
//Function Feature: transmit data in SPI mode of PIC microcontroller
//Input Argument: data: bytes of data to be transmitted
//Output Argument: void
//**************************************************************************************************
void SPI_DataSend(const unsigned char data)
{
    SSPBUF = data;    //Start sending
    while(!SSPIF);    //Wait for data being sent
    SSPIF = 0;        //Clear flag
}

//**************************************************************************************************
//Function Name: PrintString
//Function Feature: up to 4 ASCII chars to be sent
//Input Argument: string: strings to be sent
//Output Argument: void
//**************************************************************************************************
void Print(void)
{
    unsigned char i = 0;
    AddressWrite(0x00);
    SPI_ModelConfigure();
    for(i=0; i<32; i++)
    {
        SPI_DataSend(0xaa);
    }
    SSPCON = 0x11;
}
```

5.3 Heat Dissipation

Following are pictures of heat dissipation gained by Fluke Ti20 Thermal Imager in the condition of info board working at full load, all LEDs on and 100% PWM duty cycle.

5.3.1 3208 Monocolor LED 3mm Dot Matrix Display Info Board

FIGURE 5-2 HEAT DISTRIBUTION OF HT1632C'S CENTER ON THE BACK PANEL

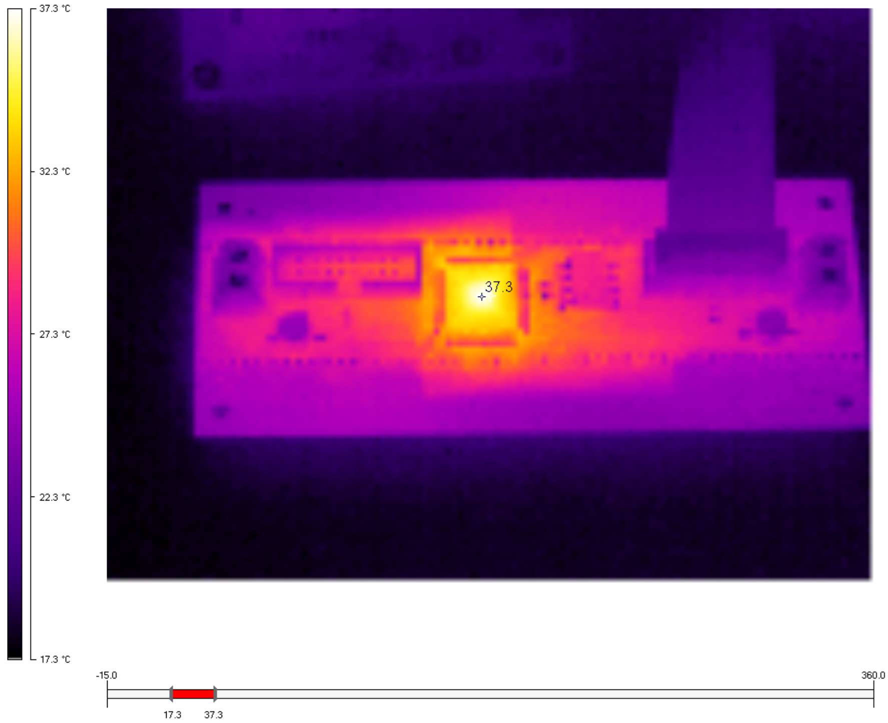
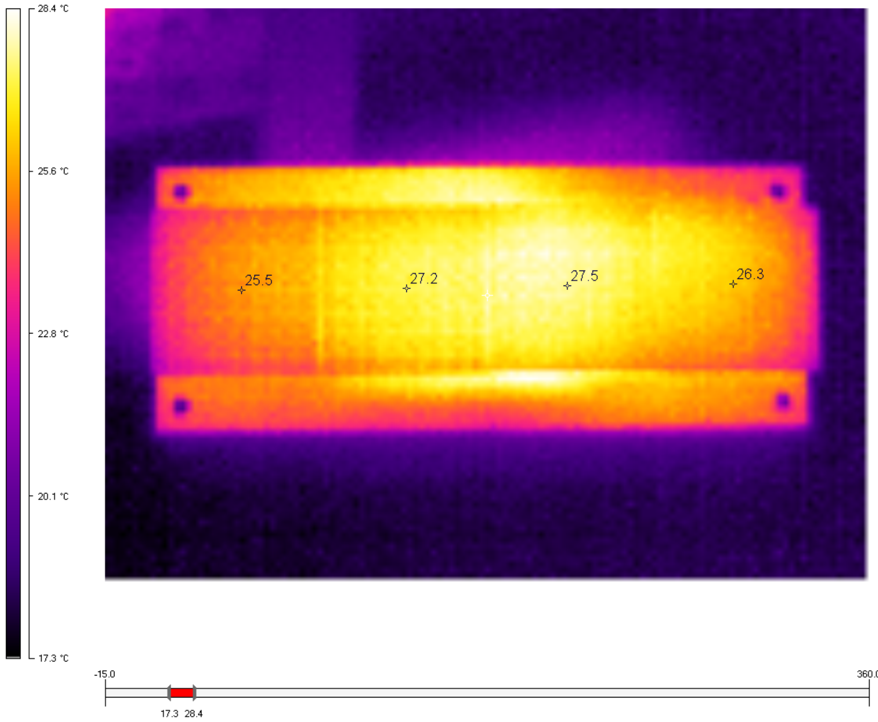


FIGURE 5-3 HEAT DISTRIBUTION OF THE FRONT PANEL



5.3.2 3208 Monocolor LED 5mm Dot Matrix Display Info Board

FIGURE 5-4 HEAT DISTRIBUTION OF HT1632C'S CENTER ON THE BACK PANEL

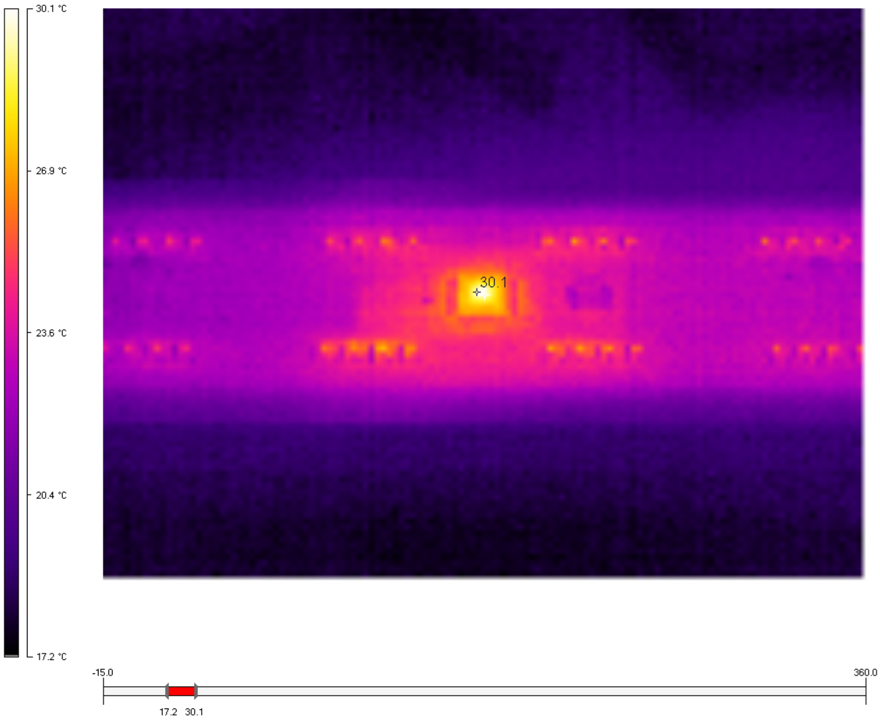
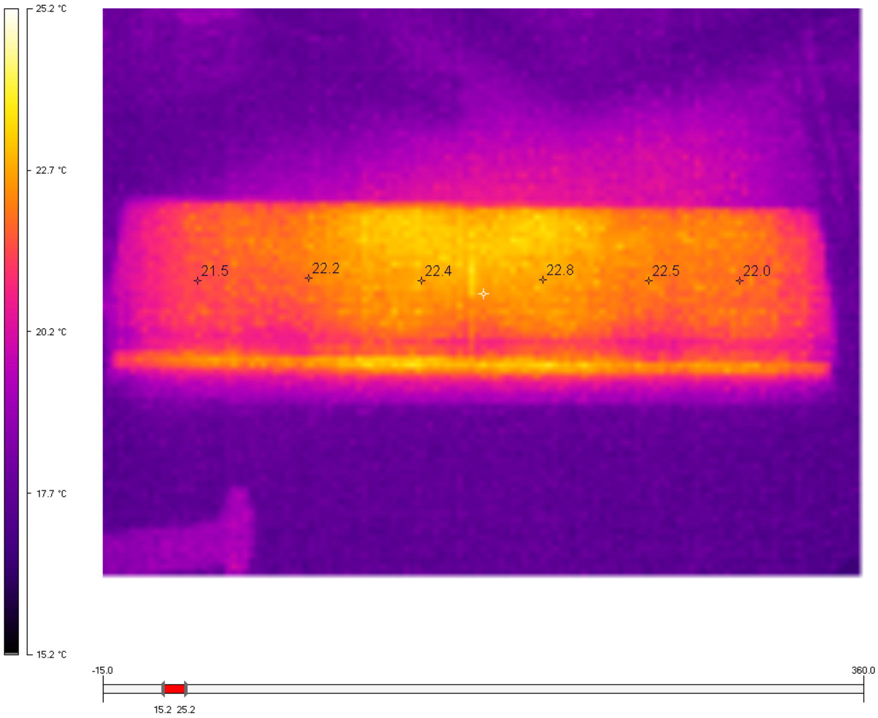


FIGURE 5-5 HEAT DISTRIBUTION OF THE FRONT PANEL



Note: As you see, these products have good heat dissipation. It's still recommended to use them in a good thermal environment.

Chapter 6. Contact Us

Sure Electronics Co., Ltd.
5F, Zone A,
Qinhuai Technology Innovation Center
105-2 DaMing Rd (ZIP:210022)
Nanjing
P.R.China
Tel: +86-13601408832 (For technical questions only)
+86-25-66606340 (English service, from GMT1-10AM)
Fax: +86-25- 66606341-866
Website: www.sure-electronics.com
www.sure-electronics.net